Abstract
Self-attention has become an important and widely used neural network component that helped to establish new state-of-the-art results for various applications, such as machine translation and automatic speech recognition (ASR). However, the computational complexity of self-attention grows quadratically with the input sequence length. This can be particularly problematic for applications such as ASR, where an input sequence generated from an utterance can be relatively long. In this work, we propose a combination of restricted self-attention and a dilation mechanism, which we refer to as dilated self-attention. The restricted self-attention allows attention to neighboring frames of the query at a high resolution, and the dilation mechanism summarizes distant information to allow attending to it with a lower resolution. Different methods for summarizing distant frames are studied, such as subsampling, mean-pooling, and attention-based pooling. ASR results demonstrate substantial improvements compared to restricted self-attention alone, achieving similar results compared to full-sequence based self-attention with a fraction of the computational costs.

IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)
CAPTURING MULTI-RESOLUTION CONTEXT BY DILATED SELF-ATTENTION

Niko Moritz, Takaaki Hori, Jonathan Le Roux

Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA, USA

ABSTRACT

Self-attention has become an important and widely used neural network component that helped to establish new state-of-the-art results for various applications, such as machine translation and automatic speech recognition (ASR). However, the computational complexity of self-attention grows quadratically with the input sequence length. This can be particularly problematic for applications such as ASR, where an input sequence generated from an utterance can be relatively long. In this work, we propose a combination of restricted self-attention and a dilation mechanism, which we refer to as dilated self-attention. The restricted self-attention allows attention to neighboring frames of the query at a high resolution, and the dilation mechanism summarizes distant information to allow attending to it with a lower resolution. Different methods for summarizing distant frames are studied, such as subsampling, mean-pooling, and attention-based pooling. ASR results demonstrate substantial improvements compared to restricted self-attention alone, achieving similar results compared to full-sequence based self-attention with a fraction of the computational costs.

Index Terms: dilated self-attention, transformer, automatic speech recognition, computational complexity

1. INTRODUCTION

The attention mechanism has become a central component in many neural network architectures for machine translation, speech processing, language modeling, and computer vision [1–5]. It is frequently used as a relay between encoder and decoder neural networks or as a substitute for recurrent neural networks (RNNs) and convolutional neural networks (CNNs) [6, 7]. Recently, the transformer architecture [8], which utilizes attention throughout all model components, has set new state-of-the-art results for many different machine learning applications, including many speech and audio processing tasks [9–12]. As attention-based architectures have been successfully applied for various domains, the number of model parameters have tended to increase to further improve results using deeper and wider architectures [13, 14]. However, in particular at inference time, computational costs must be kept small to enable scalability of a model for cloud-based services that serve many users simultaneously or to allow local computation on mobile devices.

The attention mechanism is a method to query information from an input sequence, which can also be regarded as reading from a memory [1]. In self-attention, each frame of an input sequence is used once as a query to read information from itself [6, 8]. In automatic speech recognition (ASR), neighboring frames of such a query frame may belong to the same phone, syllable, or word, where detailed information is required to recognize their coherency. On the other hand, distant information is required to recognize the context of sounds and words in an utterance and to adapt to speaker or recording characteristics, which typically requires less fine-grained information. This line of arguments is also applicable to other tasks, e.g., to machine translation or language modeling, where close-by words are more likely to have a dependent relationship, while only a few distant words or word groups are relevant to trace the semantic context and syntax of a sentence [15].

This hypothesis is investigated in this work by combining restricted (or time-restricted) self-attention with a dilation mechanism, whereby a high self-attention resolution for neighboring frames and a lower self-attention resolution for distant information are achieved. The proposed method, named dilated self-attention in analogy to dilated convolution [16], alleviates the quadratic computational cost growth of self-attention with the input sequence length. Various frame rate reduction methods are studied for the dilation mechanism, including subsampling as well as pooling methods that extract or compress the relevant information within a chunk of frames. We compare mean-pooling to a newly proposed attention-based pooling approach in this work. In such a setup, the computational cost of the restricted self-attention grows only linearly with the input sequence length, and the computational costs for attending to a dilation sequence are smaller by a factor M compared to standard self-attention, where M denotes the subsampling or the chunk size of the pooling operation. Thus, the overall complexity of dilated self-attention is significantly reduced, while the full context of an input sequence is still captured with different resolutions.

ASR results are reported for two different data sets using a transformer-based end-to-end ASR system. It is shown that dilated self-attention can reduce self-attention costs by several orders with almost no performance degradation for offline and streaming ASR applications. In addition, dilated self-attention demonstrates clear advances in terms of word error rates (WERs) over restricted self-attention alone. An attention-based pooling method is proposed that uses learned query vectors to compute the weighted average of each chunk by attention, where optionally a post-processing stage can be applied, e.g., to merge outputs of multiple attention heads. Among the tested pooling methods, attention-based pooling with post-processing is shown to achieve the highest robustness.

2. SYSTEM ARCHITECTURE

In this work, a joint connectionist temporal classification (CTC) and transformer-based end-to-end ASR system is used, which combines the advantages of both model types for training and decoding [10, 17] achieving state-of-the-art ASR results [18] and enabling streaming recognition of encoder-decoder based ASR systems [11, 19].

The transformer model leverages two different attention types: encoder-decoder attention and self-attention [8]. Encoder-decoder attention uses a decoder state as a query for attending to an input sequence, the encoder output. In self-attention, the queries are computed from the same input sequence, which results in an output sequence of the same length. Both attention types of the transformer model are based on the scaled dot-product attention mechanism,

\[
\text{Attention}(Q, K, V) = \text{Softmax} \left( \frac{QK^\top}{\sqrt{d_k}} \right) V, \quad (1)
\]
where \( Q \in \mathbb{R}^{n_q \times d_q}, K \in \mathbb{R}^{n_k \times d_k}, \) and \( V \in \mathbb{R}^{n_v \times d_v} \) are the queries, keys, and values, where the \( d_j \) denote dimensions and the \( n_j \) denote sequence lengths, \( d_q = d_k = d_v = n_q = n_k = n_v \) [8]. Instead of using a single attention head, multiple attention heads are used with

\[
\text{MHA}(\hat{Q}, \hat{K}, \hat{V}) = \text{Concat}(\text{Head}_1, \ldots, \text{Head}_{d_\text{head}})W^H
\]

and \( \text{Head}_i = \text{Attention}(\hat{Q}W^Q_i, \hat{K}W^K_i, \hat{V}W^V_i) \),

where \( \hat{Q}, \hat{K}, \) and \( \hat{V} \) are inputs to the multi-head attention (MHA) layer, \( \text{Head}_i \) represents the output of the \( i \)-th attention head for a total number of \( d_\text{head} \) heads, \( W^Q_i \in \mathbb{R}^{d_\text{model} \times d_q}, W^K_i \in \mathbb{R}^{d_\text{model} \times d_k}, W^V_i \in \mathbb{R}^{d_\text{model} \times d_v} \) as well as \( W^H \in \mathbb{R}^{d_k \times d_v \times d_\text{model}} \) are trainable weight matrices with typically \( d_k = d_v = d_\text{model}/d_h, \) and \( \text{Concat}_i \) denotes concatenation along the feature dimension of size \( d_v \).

The transformer encoder architecture consists of a two-layer CNN module \( \text{ENCNN} \) and a stack of self-attention based layers \( \text{ENC}_S \):

\[
X_0 = \text{ENCNN}(X) + \text{PE},
\]

\[
X_E = \text{ENC}_S(X_0),
\]

where \( \text{PE} \) are sinusoidal positional encodings [8] and \( X \) denotes a sequence of acoustic input features, which are 80-dimensional mel-spectral energies plus 3 extra features for pitch information [20]. Both CNN layers of \( \text{ENCNN} \) use a stride of size 2, a kernel size of \( 3 \times 3 \), and a ReLU activation function, which reduces the output frame rate by a factor of 4. The \( \text{ENC}_S \) module of (5) consists of \( E \) layers, where the \( e \)-th layer, for \( e = 1, \ldots, E \), is a composite of a multi-head self-attention layer

\[
X'_e = X_{e-1} + \text{MHA}_e(X_{e-1}, X_{e-1}, X_{e-1}),
\]

and two feed-forward neural networks of inner dimension \( d_{ff} \) and outer dimension \( d_\text{model} \) that are separated by a ReLU activation function as follows:

\[
X_e = X'_e + \text{FF}_e(X'_e),
\]

\[
\text{with } \text{FF}_e(X'_e) = \text{ReLU}(X'_eW^H_{e,1} + b^H_{e,1})W^H_{e,2} + b^H_{e,2},
\]

where \( W^H_{e,1} \in \mathbb{R}^{d_\text{model} \times d_{ff}}, W^H_{e,2} \in \mathbb{R}^{d_{ff} \times d_\text{model}}, b^H_{e,1} \in \mathbb{R}^{d_{ff}}, \) and \( b^H_{e,2} \in \mathbb{R}^{d_\text{model}} \) are trainable weight matrices and bias vectors.

The transformer objective function is defined as

\[
p_{\text{attn}}(Y|X_E) = \prod_{l=1}^{L} p(y_l|y_{1:l-1}, X_E)
\]

with label sequence \( Y = (y_1, \ldots, y_L) \), label subsequence \( y_{1:l-1} = (y_1, \ldots, y_{l-1}) \), and the encoder output sequence \( X_E = (x_1^E, \ldots, x_L^E) \). The term \( p(y_l|y_{1:l-1}, X_E) \) represents the transformer decoder model, which consists of a stack of \( D \) layers each applying two MHA layers, one performing self-attention using \( y_{1:l-1} \) as an input and one for encoder-decoder attention, followed by a feed-forward neural network similar to (8). Finally, the posterior probabilities are estimated by applying a fully-connected neural network to the last decoder layer and a softmax distribution over that output.

The transformer model is trained jointly with the CTC objective function \( p_{\text{ctc}} \) [18, 21] using the multi-objective loss function

\[
\mathcal{L} = -\gamma \log p_{\text{attn}} + (1 - \gamma) \log p_{\text{ctc}},
\]

where hyperparameter \( \gamma \) controls the weighting between the two objective functions \( p_{\text{attn}} \) and \( p_{\text{ctc}} \).

3. DILATED SELF-ATTENTION

Self-attention, restricted self-attention, and dilated self-attention are illustrated in Fig. 1. For restricted self-attention, the past and future context (blue frames) relative to a current query vector (green frame) is limited by using a fixed number of look-back and look-ahead frames. For dilated self-attention, restricted self-attention is combined with a dilation mechanism, which is shown in Fig. 1c. The dilation mechanism subsamples or compares the keys/values and appends the generated dilation sequence, which is of lower frame rate than the input sequence, to the restricted keys/values for the restricted self-attention process.

3.1. Dilation Mechanisms

The dilation mechanism at the \( e \)-th encoder layer first splits the keys

\[
K_i = X_{e-1} W^K_i = (k_1, \ldots, k_N)
\]

and values \( V_i = X_{e-1} W^V_i = (v_1, \ldots, v_N) \) each of length \( N \), cf. (3), into \( L = \lceil \frac{N}{M} \rceil \) non-overlapping key chunks \( C^K_{i,l} \) and value chunks \( C^V_{i,l} \) each of length \( M \), such that

\[
C^K_{i,l} = (k_{(l-1)+1}, \ldots, k_{(l-1)+M}),
\]

\[
C^V_{i,l} = (v_{(l-1)+1}, \ldots, v_{(l-1)+M}),
\]

for \( l = 1, \ldots, L \), where the last chunks, \( C^K_{i,L} \) and \( C^V_{i,L} \), are zero-padded if they have fewer than \( M \) frames. Next, subsampling or pooling techniques are applied to each chunk, to generate dilation sequences \( \Delta^K_i \) and \( \Delta^V_i \) that are appended to the restricted keys and values, respectively, by modifying (3) as follows:

\[
\text{Head}_{i,n,e} = \text{Attention}(x_{e,i}^{n-1} W^Q_i, \tilde{K}_{i,n,e}, \tilde{V}_{i,n,e})
\]

with \( \tilde{K}_{i,n,e} = \text{Concat}(k_{n-\nu_b,n+\nu_a}, \Delta^K_i) \)

and \( \tilde{V}_{i,n,e} = \text{Concat}(v_{n-\nu_b,n+\nu_a}, \Delta^V_i) \)

for \( n = 1, \ldots, N \), where \( \nu_b \) and \( \nu_a \) denote the number of look-back and look-ahead frames for the time-restriction, which corresponds to a window size of \( R = \nu_b + \nu_a + 1 \), and Concat denotes concatenation along the time dimension (frames).

The subsampling-based dilation mechanism selects the first frame of each chunk to form the dilation sequences \( \Delta^K_i = (k_1, \ldots, k_{M(l-1)+1}) \) and \( \Delta^V_i = (v_1, \ldots, v_{M(l-1)+1}) \). Alternatively to subsampling, pooling methods
can be applied to summarize the frames of each chunk. In this work, we compare three different pooling mechanisms: 1) mean-pooling (MP), 2) attention-based pooling (AP), and 3) attention-based pooling with post-processing (AP+PP).

For the mean-pooling (MP) based dilation mechanism, frames in each chunk are averaged to the mean vectors for \( l = 1, \ldots, L \), where the notation \([V,K]\) denotes the processing of either the values or the keys. We will continue to use this notation for the following equations. The sequence of mean vectors is used to form the dilation sequences \( \Delta_{l}^{[V,K]} = (\mu_{l,1}^{[V,K]}, \ldots, \mu_{l,L}^{[V,K]}) \).

In attention-based pooling (AP), we train embedding vectors to query summary information from the key and value chunks by using the attention mechanism as follows:

\[
\text{Attention}(\hat{q}_{l}^{[A,V,K]}), C_{l}^{[A,V,K]}, \Delta_{l}^{[V,K]}), \quad \text{for } \quad l = 1, \ldots, L, \quad \text{where} \quad \hat{q}_{l}^{[A,V,K]} \text{ and } C_{l}^{[A,V,K]} \text{ maps the attention head number } b = 1, \ldots, B \text{ to trainable values of dimension } d_{k}, \text{ and } B \text{ denotes the total number of attention heads.}
\]

The attention outputs \( \mu_{l,b}^{[V,K]} \) are averaged along \( b = 1, \ldots, B \) to form the dilation sequences \( \Delta_{l}^{[V,K]} = (\mu_{l,1}^{[V,K]}, \ldots, \mu_{l,L}^{[V,K]} \).

Post-processing (PP) can be applied to \( \mu_{l,b}^{[V,K]} \) to further process the AP output and to effectively output the dilation vectors using a two-layer feed-forward neural network of inner dimension \( d_{a} \) and outer dimension \( d_{a,k} \):

\[
\text{ReLU}(a_{l,b}^{[V,K]} W_{2}^{[V,K]} + b_{2}^{[V,K]}), \quad \text{for } \quad l = 1, \ldots, L, \quad \text{where} \quad W_{1}^{[V,K]} \in \mathbb{R}^{d_{a,k} \times d_{a}}, \quad W_{2}^{[V,K]} \in \mathbb{R}^{d_{a} \times \text{dim}(a_{l,b})}, \quad b_{2}^{[V,K]} \in \mathbb{R}^{d_{a}}, \quad \text{and} \quad \text{Concat}(\cdot) \text{ denotes concatenation of the vectors } \mu_{l,b}^{[V,K]} \text{ for } b = 1, \ldots, B \text{ along the feature dimension.}
\]

The complexity \( \xi \) of the AP mechanism amounts to \( \mathcal{M}(Nd_{a}d_{k}B) \) for the dot-product attention of the learned queries \( \bar{q} \) and the key chunks \( C_{l}^{[V,K]} \), where the computed attention weights are reused to summarize the value chunks \( C_{l}^{[V,K]} \) as well. The complexity of PP amounts to \( \mathcal{M}(2B + 1d_{a}d_{k}d_{a,k}d_{a} + M_{1}) \) for post-processing the attention results of the key and value chunks as described in (20).

In order to reduce the computational complexity for the following experiments, the feed-forward neural network of the post-processing stage uses a bottleneck of inner dimension \( d_{a} = 16 \).

### 3.3. Related Work

The closest prior art to this work is Transformer-XL [22] and the recently proposed self-attention with augmented memory [23]. However, both are different in various ways, e.g., Transformer-XL incorporates history information at each layer only from the previous chunk and from a lower layer. Self-attention with augmented memory uses mean-pooling on chunks of the input sequence to form a summarization query for computing attention weights over the current chunk plus the summarization outputs from previous chunks. This is a recursive operation, which cannot be easily parallelized, unlike our proposed solution. Moreover, we use learned embedding vectors for the summarization queries instead of mean-pooling, where the characteristics of the relevant frames are learned and which also allows us to use multiple queries per chunk.

### 4. EXPERIMENTS

#### 4.1. Datasets

The LibriSpeech corpus and the Wall Street Journal (WSJ) corpus are used in this work [24, 25]. LibriSpeech is a corpus of read English audio books with about 960 hours of training data, 10.7 hours of development data, and 10.5 hours of test data. The development and test data sets are both split into approximately two halves named “clean” and “other”, based on the quality of the recorded speech utterances as assessed using an ASR system [24]. WSJ is a data set of read English newspapers with approximately 81 hours of training data, 1.1 hours of development data, and 0.7 hours of test data [25]. The average duration of a LibriSpeech utterance amounts to 12.3 seconds, the median duration to 13.8 seconds, and the maximum duration to 29.7 seconds. For the WSJ corpus, the average duration of an utterance is 7.8 seconds, the median duration is 7.7 seconds, and the maximum duration is 24.4 seconds.

#### 4.2. Settings

For LibriSpeech-based experiments, the transformer model parameters are \( d_{model} = 512 \), \( d_{ff} = 2048 \), \( d_{h} = 8 \), \( E = 12 \), and \( D = 6 \). They are the same for WSJ except \( d_{model} = 256 \) and \( d_{h} = 4 \). The Adam optimizer with \( \beta_{1} = 0.9, \beta_{2} = 0.98, \epsilon = 10^{-9} \) and learning rate scheduling similar to [8] is applied for training using 25000 warmup steps. The learning rate factor and the maximum number of training epochs are set to 5.0 and 120 for LibriSpeech, and to 10.0 and 100 for WSJ. Output tokens for the LibriSpeech model consist of 5000 subwords obtained by the SentencePiece method [26]. For WSJ, the number of output tokens amounts to 50, the number of characters in WSJ. Weight factor \( \gamma \), which is used to balance the CTC and transformer model objectives during training, is set to 0.3. Layer normalization is applied before and after dropout with a rate of 10% after each MHA and FF layer. In addition, label smoothing with a penalty of 0.1 is used and SpecAugment is applied for the LibriSpeech-based ASR experiments [27]. An RNN-based language model (LM) is employed via shallow fusion for each data sets: a word-based LM of 65k words is applied for WSJ [28] and a
based pooling (AP) with and without post-processing (PP) are investigated. In addition, subsampling, mean-pooling (MP), and attention-based pooling leading to the best ASR results. Dilated self-attention can compensate to a large extent this loss in accuracy, achieving consistently better WERs compared to restricted self-attention. The overall best performing dilation mechanism is $\text{AP}$, achieving similar results compared to full-sequence based self-attention.

Table 1 shows the WSJ-based ASR results for full-sequence based self-attention, restricted self-attention, and dilated self-attention. $R$ denotes the symmetric restriction window size and $M$ the chunk size in number of frames. $M$ denotes the number of multiplications for self-attention with $N = 195$ (average input length) and $d_{\text{input}} = 256$. Numbers after a hyphen denote the number of attention heads for AP.

<table>
<thead>
<tr>
<th>self-attention type</th>
<th>dilation mechanism</th>
<th>$R$</th>
<th>$M$</th>
<th>$\mathcal{M}$</th>
<th>dev</th>
<th>test</th>
</tr>
</thead>
<tbody>
<tr>
<td>full-sequence</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>9.8M</td>
<td>7.7</td>
<td>4.7</td>
</tr>
<tr>
<td>restricted</td>
<td>-</td>
<td>35</td>
<td>1.8M</td>
<td>8.7</td>
<td>5.9</td>
<td></td>
</tr>
<tr>
<td>restricted</td>
<td>-</td>
<td>21</td>
<td>1.1M</td>
<td>8.7</td>
<td>5.6</td>
<td></td>
</tr>
<tr>
<td>restricted</td>
<td>-</td>
<td>15</td>
<td>0.8M</td>
<td>8.6</td>
<td>5.8</td>
<td></td>
</tr>
<tr>
<td>restricted</td>
<td>-</td>
<td>13</td>
<td>0.7M</td>
<td>9.0</td>
<td>5.7</td>
<td></td>
</tr>
<tr>
<td>restricted</td>
<td>-</td>
<td>11</td>
<td>0.6M</td>
<td>8.9</td>
<td>6.2</td>
<td></td>
</tr>
<tr>
<td>dilated subsampling</td>
<td>15</td>
<td>10</td>
<td>1.8M</td>
<td>8.0</td>
<td>5.2</td>
<td></td>
</tr>
<tr>
<td>dilated MP</td>
<td>15</td>
<td>10</td>
<td>1.8M</td>
<td>7.9</td>
<td>5.4</td>
<td></td>
</tr>
<tr>
<td>dilated AP-1</td>
<td>15</td>
<td>11</td>
<td>1.8M</td>
<td>7.7</td>
<td>4.9</td>
<td></td>
</tr>
<tr>
<td>dilated AP-2</td>
<td>15</td>
<td>12</td>
<td>1.8M</td>
<td>7.7</td>
<td>4.9</td>
<td></td>
</tr>
<tr>
<td>dilated AP-1+PP</td>
<td>21</td>
<td>20</td>
<td>1.8M</td>
<td>7.7</td>
<td>4.9</td>
<td></td>
</tr>
<tr>
<td>dilated AP-2+PP</td>
<td>21</td>
<td>27</td>
<td>1.8M</td>
<td>7.6</td>
<td>5.1</td>
<td></td>
</tr>
<tr>
<td>dilated subsampling</td>
<td>11</td>
<td>20</td>
<td>1.1M</td>
<td>8.1</td>
<td>5.3</td>
<td></td>
</tr>
<tr>
<td>dilated MP</td>
<td>11</td>
<td>20</td>
<td>1.1M</td>
<td>7.7</td>
<td>5.2</td>
<td></td>
</tr>
<tr>
<td>dilated AP-1</td>
<td>11</td>
<td>22</td>
<td>1.1M</td>
<td>7.6</td>
<td>5.1</td>
<td></td>
</tr>
<tr>
<td>dilated AP-2</td>
<td>11</td>
<td>28</td>
<td>1.1M</td>
<td>7.7</td>
<td>5.1</td>
<td></td>
</tr>
<tr>
<td>dilated AP-1+PP</td>
<td>9</td>
<td>24</td>
<td>1.1M</td>
<td>8.2</td>
<td>5.2</td>
<td></td>
</tr>
<tr>
<td>dilated AP-2+PP</td>
<td>9</td>
<td>33</td>
<td>1.1M</td>
<td>7.6</td>
<td>4.9</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>self-attention type</th>
<th>dilation mechanism</th>
<th>$R$</th>
<th>$M$</th>
<th>$\mathcal{M}$</th>
<th>dev</th>
<th>test</th>
</tr>
</thead>
<tbody>
<tr>
<td>restricted</td>
<td>-</td>
<td>-</td>
<td>2.8</td>
<td>7.5</td>
<td>3.1</td>
<td>8.1</td>
</tr>
<tr>
<td>dilated</td>
<td>AP-2+PP</td>
<td>9</td>
<td>1</td>
<td>2.9</td>
<td>7.9</td>
<td>3.0</td>
</tr>
</tbody>
</table>

Table 2. LibriSpeech-based WER [%] for self-attention, restricted self-attention, and dilated self-attention with results for streaming end-to-end ASR at the bottom [11]. The computational complexity estimate $\mathcal{M}$ is based on using $d_{\text{input}} = 512$ and $N = 310$, which is the average encoder sequence length of a LibriSpeech utterance.

<table>
<thead>
<tr>
<th>self-attention type</th>
<th>dilation mechanism</th>
<th>$R$</th>
<th>$M$</th>
<th>$\mathcal{M}$</th>
<th>dev</th>
<th>test</th>
</tr>
</thead>
<tbody>
<tr>
<td>full-sequence</td>
<td>-</td>
<td>-</td>
<td>52M</td>
<td>2.3</td>
<td>5.7</td>
<td>2.6</td>
</tr>
<tr>
<td>restricted</td>
<td>-</td>
<td>25</td>
<td>4.0M</td>
<td>2.4</td>
<td>6.5</td>
<td>7.0</td>
</tr>
<tr>
<td>restricted</td>
<td>-</td>
<td>13</td>
<td>2.1M</td>
<td>2.5</td>
<td>7.0</td>
<td>7.2</td>
</tr>
<tr>
<td>dilated subsampling</td>
<td>25</td>
<td>20</td>
<td>6.5M</td>
<td>2.3</td>
<td>6.1</td>
<td>6.2</td>
</tr>
<tr>
<td>dilated MP</td>
<td>25</td>
<td>20</td>
<td>6.5M</td>
<td>2.3</td>
<td>6.1</td>
<td>6.2</td>
</tr>
<tr>
<td>dilated AP-1</td>
<td>25</td>
<td>20</td>
<td>6.5M</td>
<td>2.3</td>
<td>6.1</td>
<td>6.2</td>
</tr>
<tr>
<td>dilated AP-2</td>
<td>25</td>
<td>20</td>
<td>6.8M</td>
<td>2.3</td>
<td>6.1</td>
<td>6.1</td>
</tr>
<tr>
<td>dilated AP-1+PP</td>
<td>25</td>
<td>20</td>
<td>7.2M</td>
<td>2.3</td>
<td>5.9</td>
<td>6.1</td>
</tr>
<tr>
<td>dilated AP-2+PP</td>
<td>25</td>
<td>20</td>
<td>7.6M</td>
<td>2.3</td>
<td>5.9</td>
<td>6.1</td>
</tr>
<tr>
<td>dilated AP-2+PP</td>
<td>17</td>
<td>19</td>
<td>6.6M</td>
<td>2.2</td>
<td>5.8</td>
<td>5.6</td>
</tr>
<tr>
<td>dilated subsampling</td>
<td>13</td>
<td>40</td>
<td>3.3M</td>
<td>2.4</td>
<td>6.5</td>
<td>6.7</td>
</tr>
<tr>
<td>dilated MP</td>
<td>13</td>
<td>40</td>
<td>3.3M</td>
<td>2.5</td>
<td>6.6</td>
<td>6.6</td>
</tr>
<tr>
<td>dilated AP-1</td>
<td>11</td>
<td>34</td>
<td>3.5M</td>
<td>2.4</td>
<td>6.0</td>
<td>2.5</td>
</tr>
<tr>
<td>dilated AP-2+PP</td>
<td>11</td>
<td>50</td>
<td>3.5M</td>
<td>2.2</td>
<td>5.9</td>
<td>2.5</td>
</tr>
</tbody>
</table>

Finally, the AP-2+PP approach is applied to a triggered attention (TA) based streaming end-to-end ASR system [11], where a TA look-ahead of 12 frames and an encoder look-ahead of 1 frame are used, which results in an algorithmic delay of 480 ms for the TA-based decoder plus 480 ms for the encoder. In this setup, the dilation mechanism only processes past encoder frames, i.e., input frames up to the query frame. Results demonstrate that dilated self-attention, where the dilation mechanism is applied to past frames only, achieves about similar ASR results compared to using the full past for self-attention, while the computational complexity is substantially reduced. For example, for computing the next self-attention output with 4 or 8 seconds of past audio input, the computational complexity of self-attention is reduced by a factor 7.2 or 11.8 if no new dilation output is computed and by a factor 1.25 or 2.4 when a new full chunk is processed for extending the dilation sequences $\Delta_i^{[V,K]}$, which is computed every $M=15$ frames.

5. CONCLUSIONS

We proposed a dilation mechanism to reduce the computational complexity of self-attention by enabling attention to an input sequence with different levels of resolution. Subsampling, mean-pooling, and attention-based pooling techniques are investigated for summarizing and capturing long-context information in self-attention and to reduce the frame rate of distant information relative to a query. Dilated self-attention is non-recursive, which allows it to be parallelized for efficient implementation, and it has demonstrated substantial improvements compared to restricted self-attention with attention-based pooling leading to the best ASR results. Dilated self-attention has reached WERs of 2.4% and 5.9% for the test-clean and test-other conditions of LibriSpeech, which is even slightly lower than the standard self-attention mode with only 15% of the computational costs for self-attention. In addition, dilated self-attention has been applied to a triggered attention-based ASR system, where it has demonstrated to be effective for streaming ASR as well.
6. REFERENCES


